Data Mining: Data Preprocessing

Prof.Dr. Songül Varlı

Department of Computer Engineering

Yildiz Technical University

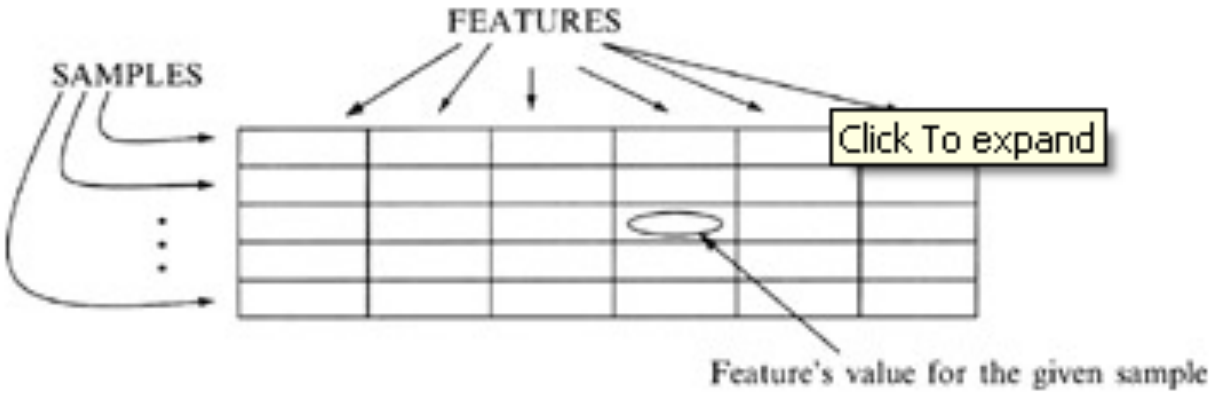
svarli@yildiz.edu.tr

Preparing the Data

- Chapter Objectives
  - Analyze basic representations and characteristics of raw and large data sets.
  - Apply different normalization techniques on numerical attributes.
  - Recognize different techniques for data preparation, including attribute transformation.
  - Compare different methods for elimination of missing values.
  - Construct a method for uniform representation of time-dependent data
  - Compare different methods for outlier detection.

REPRESENTATION OF RAW DATA

- Data samples introduced as rows in table below are basic components in a data mining process.
- Every sample is described with several features and there are different types of values for every feature.



REPRESENTATION OF RAW DATA

- We will start with the two most common data types:
  - numeric
  - categorical

REPRESENTATION OF RAW DATA: Numeric Data

- Numeric values include real-value variables or integer variables such as age, speed, or length.

- A feature with numeric values has two important properties:
  - its values have an **order relation** (2 < 5 and 5 < 7) and
  - a **distance relation** (d(2.3, 4.2) = 1.9).

REPRESENTATION OF RAW DATA: Categoric Data

- In contrast, categorical (often called symbolic) variables have neither of these two relations. The two values of a categorical variable can be either equal or not equal: they only support an equality relation (Blue = Blue, or Red ≠ Black).
- Examples of variables of this type are eye color, sex, or country of citizenship.

REPRESENTATION OF RAW DATA: Categoric Data

- A categorical variable with two values can be converted, in principle, to a numeric binary variable with two values: 0 or 1.
- A categorical variable with N values can be converted into N binary numeric variables, namely, one binary variable for each categorical value. These coded categorical variables are known as "dummy variables" in statistics.
- For example, if the variable eye-color has four values: black, blue, green, and brown, they can be coded with four binary digits.

Feature value	Code
Black	1000
Blue	0100
Green	0010
Brown	0001

REPRESENTATION OF RAW DATA

- Another way of classifying variable, based on its values, is to look at it as a continuous variable or a discrete variable.

REPRESENTATION OF RAW DATA: Continuous Variables

- Continuous variables are also known as quantitative or metric variables. They are measured using either an interval scale or a ratio scale.
- Both scales allow the underlying variable to be defined or measured theoretically with infinite precision.
- The difference between these two scales lies in how the zero point is defined in the scale.
- The zero point in the interval scale is placed arbitrarily and thus it does not indicate the complete absence of whatever is being measured. The best example of the interval scale is the temperature scale, where zero degrees Fahrenheit does not mean a total absence of temperature. Because of the arbitrary placement of the zero point, the ratio relation does not hold true for variables measured using interval scales. For example, 80 degrees Fahrenheit does not imply twice as much heat as 40 degrees Fahrenheit.
- In contrast, a ratio scale has an absolute zero point and, consequently, the ratio relation holds true for variables measured using this scale. Quantities such as height, length, and salary use this type of scale.
- Continuous variables are represented in large data sets with values that are numbers-real or integers.

REPRESENTATION OF RAW DATA: Discrete Variables

- Discrete variables are also called qualitative variables. Such variables are measured, or its values defined, using one of two kinds of non metric scales-nominal or ordinal.
- A nominal scale is an orderless scale, which uses different symbols, characters, and numbers to represent the different states (values) of the variable being measured.
  - An example of a nominal variable, a utility, customer-type identifier with possible values is residential, commercial, and industrial. These values can be coded alphabetically as A, B, and C, or numerically as 1, 2. or 3, but they do not have metric characteristics as the other numeric data have.
  - Another example of a nominal attribute is the zip-code field available in many data sets. In both examples, the numbers used to designate different attribute values have no particular order and no necessary relation to one another.

REPRESENTATION OF RAW DATA: Discrete Variables

- An ordinal scale consists of ordered, discrete gradations, e.g., rankings.
- An ordinal variable is a categorical variable for which an order relation is defined but not a distance relation.
  - Some examples of an ordinal attribute are the rank of a student in a class and the gold, silver, and bronze medal positions in a sports competition. The ordered scale need not be necessarily linear; e.g., the difference between the students ranked 4th and 5th need not be identical to the difference between the students ranked 15th and 16th. All that can be established from an ordered scale for ordinal attributes is greater-than, equal-to, or less-than relations.
  - Typically, ordinal variables encode a numeric variable onto a small set of overlapping intervals corresponding to the values of an ordinal variable. These ordinal variables are closely related to the linguistic or fuzzy variables commonly used in spoken English;
    - e.g., AGE (with values young, middle-aged, and old) and
    - INCOME (with values low, middle-class, upper-middle-class, and rich).

CHARACTERISTICS OF RAW DATA

- All raw data sets initially prepared for data mining are often large; many are related to human beings and have the potential for being messy.
- A priori, one should expect to find missing values, distortions, misrecording, inadequate sampling, and so on in these initial data sets.
- Raw data that do not appear to show any of these problems should immediately arouse suspicion.
- The only real reason for the high quality of data could be that the presented data have been cleaned up and preprocessed before the analyst sees them, as in data of a correctly designed and prepared data warehouse.

CHARACTERISTICS OF RAW DATA : what the sources and implications of messy data

- First, data may be missing for a huge variety of reasons.
- Sometimes there are mistakes in measurements or recordings, but in many cases, the value is unavailable.
- To cope with this in a data-mining process, one must not only be able to model with the data that are presented, but even with their values missing.
- We will see later that some data mining techniques are more or less sensitive to missing values. If the method is robust enough, then the missing values are not a problem. Otherwise, it is necessary to solve the problem of missing values before the application of a selected data-mining technique.

CHARACTERISTICS OF RAW DATA : what the sources and implications of messy data

- The second cause of messy data is misrecorded data, and that is typical in large volumes of data.
- We have to have mechanisms to discover some of these "unusual" values; in some cases, even to work with them to eliminate their influence on the final results.
- Further, data may not be from the population they are supposed to be from. Outliers are typical examples here, and they require careful analysis before the analyst can decide whether they should be dropped from the data-mining process as anomalous or included as unusual examples from the population under study.

TRANSFORMATION OF RAW DATA

- We will review a few general types of transformations of data that are not problem-dependent and that may improve data-mining results.
- Selection of techniques and use in particular applications depend on types of data, amounts of data, and general characteristics of the data-mining task.

TRANSFORMATION OF RAW DATA
1-Normalizations

- Some data-mining methods, typically those that are based on distance computation between points in an n-dimensional space, may need normalized data for best results.
- The measured values can be scaled to a specific range, e.g., [−1, 1], or [0, 1].
- If the values are not normalized, the distance measures will overweight those features that have, on an average, larger values.
- There are many ways of normalizing data. Here are three simple and effective normalization techniques:
  - **Decimal scaling**
  - **Min-max normalization**
  - **Standard deviation normalization**

TRANSFORMATION OF RAW DATA
1-Normalizations

Decimal Scaling:

- Decimal scaling moves the decimal point but still preserves most of the original digit value.
- The typical scale maintains the values in a range of −1 to 1.
- The following equation describes decimal scaling, where v(i) is the value of the feature v for case i and v′(i) is a scaled value

$$v'(i) = v(i) / (10^k)$$

- for the smallest k such that max (|v′(i)|) < 1.
- First the maximum |v′(i)| is found in the data set and then, the decimal point is moved until the new, scaled, maximum absolute value is less than 1. The divisor is then applied to all other v(i).
- For example, if the largest value in the set is 455 and the smallest value is −834, then the maximum absolute value of the feature becomes .834, and the divisor for all v(i) is 1000 (k = 3).

TRANSFORMATION OF RAW DATA
1-Normalizations

Min-max normalization:

- Suppose that the data for a feature v are in a range between 150 and 250. Then, the previous method of normalization will give all normalized data between .15 and .25; but it will accumulate the values on a small subinterval of the entire range. To obtain better distribution of values on a whole, normalized interval, e.g., [0, 1], we can use the min-max formula

$$v'(i) = (\ v(i) - \min(v(i)) \ ) \ / \ (\ \max(v(i)) \ - \min(v(i)) \ )$$

- where the minimum and the maximum values for the feature v are computed on a set automatically, or they are estimated by an expert in a given domain.
- Similar transformation may be used for the normalized interval [−1, 1]. The automatic-computation of min and max values requires one additional search through the entire data set, but, computationally, the procedure is very simple. On the other hand, expert estimations of min and max values may cause unintentional accumulation of normalized values.

TRANSFORMATION OF RAW DATA
1-Normalizations

Standard deviation normalization:

- Normalization by standard deviation often works well with distance measures, but transforms the data into a form unrecognizable from the original data.
- For a feature v, the mean value mean(v) and the standard deviation sd(v) are computed for the entire data set. Then, for a case i, the feature value is transformed using the equation

$$v'(i) = (v(i) - \text{mean}(v)) \ / \ \text{sd}(v)$$

- For example, if the initial set of values of the attribute is v = {1, 2, 3}, then mean(v) = 2, sd(v) = 1, and the new set of normalized values is v* = {−1, 0, 1}.

TRANSFORMATION OF RAW DATA
2-Data smoothing

- A numeric feature, y, may range over many distinct values, sometimes as many as the number of training cases.
- For many data-mining techniques, minor differences between these values are not significant and may degrade the performance of the method and the final results.
- They may be considered as random variations of the same underlying value. Hence, it can be advantageous sometimes to smooth the values of the variable.
- Many simple smoothers can be specified that average similar measured values. For example, if the values are real numbers with several decimal places, rounding the values to the given precision could be a simple smoothing algorithm for a large number of samples, where each sample has its own real value. If the set of values for the given feature F is {0.93, 1.01, 1.001, 3.02, 2.99, 5.03, 5.01, 4.98}, then it is obvious that smoothed values will be Fsmoothed = {1.0, 1.0, 1.0, 3.0, 3.0, 5.0, 5.0, 5.0}.
- This simple transformation is performed without losing any quality in a data set, and, at the same time, it reduces the number of different real values for the feature to only three.

TRANSFORMATION OF RAW DATA
2-Data smoothing

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by human (e.g., deal with possible outliers)

TRANSFORMATION OF RAW DATA
2-Data smoothing: Simple Discretization Methods (Binning)

- Equal-width (distance) partitioning
  - Divides the range into N intervals of equal size: uniform grid
  - if A and B are the lowest and highest values of the attribute, the width of intervals will be: W = (B –A)/N.
  - The most straightforward, but outliers may dominate presentation
  - Skewed data is not handled well
- Equal-depth (frequency) partitioning
  - Divides the range into N intervals, each containing approximately same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky

TRANSFORMATION OF RAW DATA
2- Binning Methods for Data Smoothing

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* Partition into equal-frequency (equi-depth) bins:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
* Smoothing by bin means:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
* Smoothing by bin boundaries:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

TRANSFORMATION OF RAW DATA
3-Differences and ratios

- Even small changes to features can produce significant improvement in data-mining performances.
- The effects of relatively minor transformations of input or output features are particularly important in the specification of the data-mining goals.
- Two types of simple transformations, differences and ratios, could make improvements in goal specification, especially if they are applied to the output features.

TRANSFORMATION OF RAW DATA
3-Differences and ratios

- Differences and ratio transformations are not only useful for output features, but also for inputs.
- They can be used as changes in time for one feature or as a composition of different input features.
- For example, in many medical data sets, there are two features of a patient, height and weight, that are taken as input parameters for different diagnostic analyses. Many applications show that better diagnostic results are obtained when an initial transformation is performed using a new feature called the body-mass index (BMI), which is the weighted ratio between weight and height.

MISSING DATA

- For many real-world applications of data mining, even when there are huge amounts of data, the subset of cases with complete data may be relatively small.
- Available samples and also future cases may have values missing.
- Some of the data-mining methods accept missing values and satisfactorily process data to reach a final conclusion.
- Other methods require that all values be available.
- An obvious question is whether these missing values can be filled in during data preparation, prior to the application of the data-mining methods.
- The simplest solution for this problem is the reduction of the data set and the elimination of all samples with missing values. That is possible when large data sets are available, and missing values occur only in a small percentage of samples. If we do not drop the samples with missing values, then we have to find values for them. What are the practical solutions?

MISSING DATA

- First, a data miner, together with the domain expert, can manually examine samples that have no values and enter a reasonable, probable, or expected value, based on a domain experience.

- The method is straightforward for small numbers of missing values and relatively small data sets. But, if there is no obvious or plausible value for each case, the miner is introducing noise into the data set by manually generating a value.

MISSING DATA

- The second approach gives an even simpler solution for elimination of missing values. It is based on a formal, often automatic replacement of missing values with some constants, such as:
  - Replace all missing values with a single global constant (a selection of a global constant is highly application-dependent).
  - Replace a missing value with its feature mean.
  - Replace a missing value with its feature mean for the given class (this approach is possible only for classification problems where samples are classified in advance).

Example of Missing Data

Case ID	V_1	V_2	V_3	V_4	V_5	*Missing Data by Case* Number	Percent
1	1.3	9.9	6.7	3.0	2.6	0	0
2	4.1	5.7			2.9	2	40
3		9.9		3.0		3	60
4	.9	8.6		2.1	1.8	1	20
5	.4	8.3		1.2	1.7	1	20
6	1.5	6.7	4.8		2.5	1	20
7	.2	8.8	4.5	3.0	2.4	0	0
8	2.1	8.0	3.0	3.8	1.4	0	0
9	1.8	7.6		3.2	2.5	1	20
10	4.5	8.0		3.3	2.2	1	20
11	2.5	9.2		3.3	3.9	1	20
12	4.5	6.4	5.3	3.0	2.5	0	9
13					2.7	4	80
14	2.8	6.1	6.4		3.8	1	20
15	3.7			3.0		3	60
16	1.6	6.4	5.0		2.1	1	20
17	.5	9.2		3.3	2.8	1	20
18	2.8	5.2	5.0		2.7	1	20
19	2.2	6.7		2.6	2.9	1	20
20	1.8	9.0	5.0	2.2	3.0	0	0
Missing Data by Variable						Total Missing Values	
Number	2	2	11	6	2	Number: 23	
Percent	10	10	55	30	10	Percent: 23	

Practical Considerations

- Complete data required
  - Only 5 cases are usable (too few)
- A possible remedy: Eliminate V_3
  - 12 cases have complete data
  - Eliminate cases 3, 13, 15
  - Total number of missing data is reduced to 7.4% for all values

TIME-DEPENDENT DATA

- Practical data-mining applications will range from those having strong time-dependent relationships to those with loose or no time relationships.
- Real-world problems with time dependencies require special preparation and transformation of data, which are, in many cases, critical for successful data mining.
- We will start with the simplest case-a single feature measured over time. This feature has a series of values over fixed time units. For example, a temperature reading could be measured every hour, or the sales of a product could be recorded every day. This is the classical univariate time-series problem, where it is expected that the value of the variable X at a given time be related to previous values. Because the time series is measured at fixed units of time, the series of values can be expressed as

  $$X = \{ t(1), t(2), t(3), \ldots, t(n) \}$$

  where t(n) is the most recent value.

TIME-DEPENDENT DATA

- For many time-series problems, the goal is to forecast t(n + 1) from previous values of the feature, where these values are directly related to the predicted value.
- One of the most important steps in preprocessing of row, time-dependent data is the specification of **a window or a time lag**. This is the number of previous values that influence the prediction. Every window represents one sample of data for further analysis. For example, if the time series consists of the eleven measurements

  X={ t(0), t(1), t(2), t(3), t(4), t(5), t(6), t(7), t(8), t(9), t(10) }

- and if the window for analysis of the time-series is five, then it is possible to reorganize the input data into a tabular form with six samples, which is more convenient (standardized) for the application of data-mining techniques. Transformed data are given in the following Table

TIME-DEPENDENT DATA

Table 2.1: Transformation of Time Series to standard tabular form (window = 5)

Sample	WINDOW					Next Value
	M1	M2	M3	M4	M5	
1	t(0)	t(1)	t(2)	t(3)	t(4)	t(5)
2	t(1)	t(2)	t(3)	t(4)	t(5)	t(6)
3	t(2)	t(3)	t(4)	t(5)	t(6)	t(7)
4	t(3)	t(4)	t(5)	t(6)	t(7)	t(7)
5	t(4)	t(5)	t(6)	t(7)	t(8)	t(9)
6	t(5)	t(6)	t(7)	t(8)	t(9)	t(10)

TIME-DEPENDENT DATA

- While the typical goal is to predict the next value in time, in some applications, the goal can be modified to predict values in the future, several time units in advance.
- More formally, given the time-dependent values t(n – i), …, t(n), it is necessary to predict the value t(n + j). In the previous example, taking j = 3, the new samples are given in the following table.

Table 2.2: Time-series samples in standard tabular form (window = 5) with postponed predictions (j = 3)

Sample	WINDOW					Next Value
	M1	M2	M3	M4	M5	
1	t(0)	t(1)	t(2)	t(3)	t(4)	t(7)
2	t(1)	t(2)	t(3)	t(4)	t(5)	t(8)
3	t(2)	t(3)	t(4)	t(5)	t(6)	t(9)
4	t(3)	t(4)	t(5)	t(6)	t(7)	t(10)

TIME-DEPENDENT DATA

- In general, the further out in the future, the more difficult and less reliable is the forecast.
- The goal for a time series can easily be changed from predicting the next value in the time series to classification into one of predefined categories.
- From a data preparation perspective there are no significant changes.
- For example, instead of predicted output value t(i + 1), the new classified output will be binary: T for t(i + 1) ≥ threshold value and F for t(i + 1) < threshold value.

OUTLIER ANALYSIS

- Very often, in large data sets, there exists samples that do not comply with the general behavior of the data model.
- Such samples, which are significantly different or inconsistent with the remaining set of data, are called outliers.
- Outliers can be caused by measurement error or they have be the result of inherent data variability.
- If, e.g., the display of a person's age in the database is −1 the value is obviously not correct, and error could have been caused by a default setting of the field "unrecorded age" in the computer program.
- On the other hand, if, in the database, the number of children for one person is 25 this datum is unusual and has to be checked. The value could be a typographical error, or it could be correct and represent real variability for the given attribute.

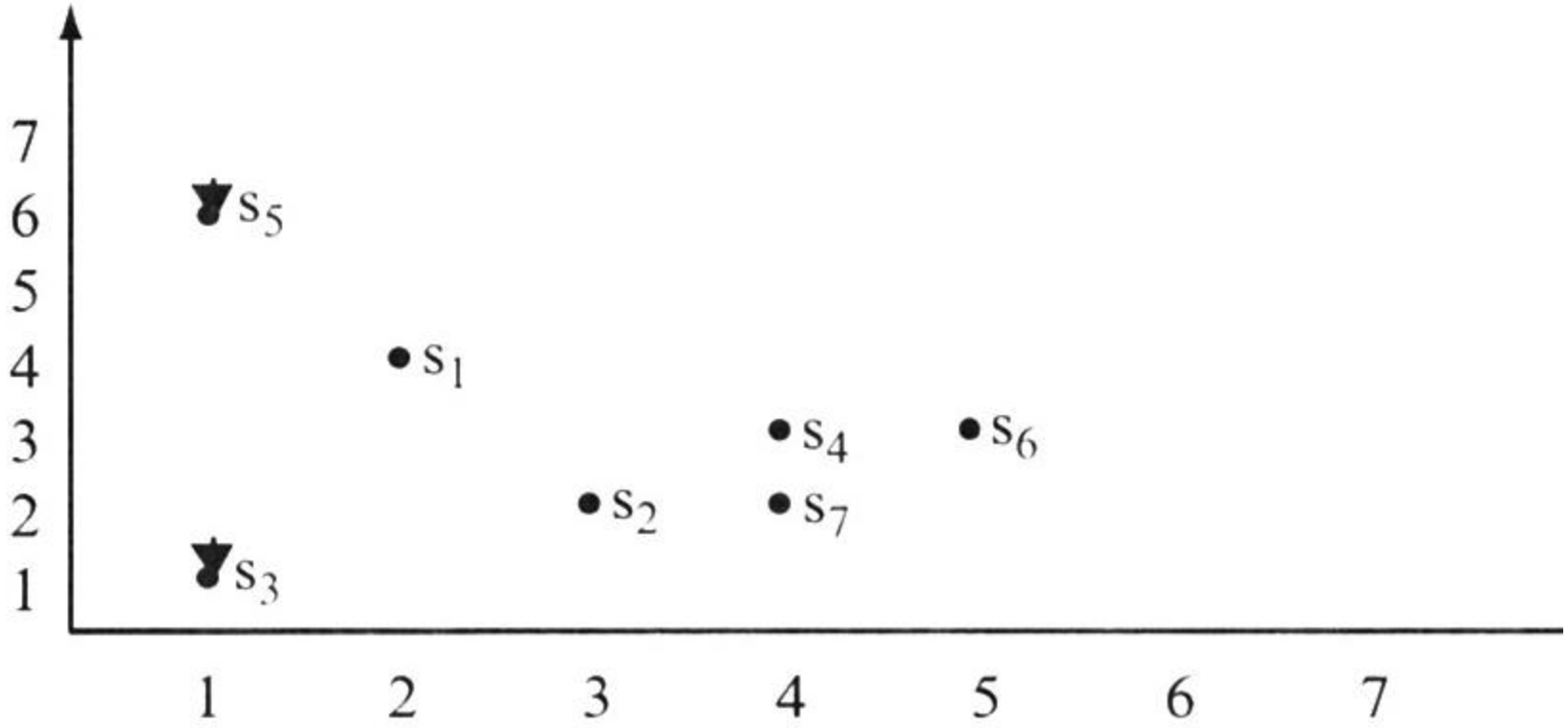
OUTLIER ANALYSIS

- Many data-mining algorithms try to minimize the influence of outliers on the final model, or to eliminate them in the preprocessing phases.
- The data-mining analyst has to be very careful in the automatic elimination of outliers because, if the data are correct, that could result in the loss of important hidden information. Some data-mining applications are focused on outlier detection, and it is the essential result of a data analysis.
- For example, while detecting fraudulent credit card transactions in a bank, the outliers are typical examples that may indicate fraudulent activity, and the entire data-mining process is concentrated on their detection.
- But, in most of the other data-mining applications, especially if they are supported with large data sets, outliers are not very useful, and they are more the result of errors in data collection then a characteristic of a data set.

OUTLIER ANALYSIS

- Outlier detection and potential removal from a data set can be described as a process of the selection of k out of a samples that are considerably dissimilar, exceptional, or inconsistent with respect to the remaining data.
- The problem of defining outliers is nontrivial, especially in multidimensional samples.
- Data visualization methods that are useful in outlier detection for one to three dimensions are weaker in multidimensional data because of a lack of adequate visualization methodologies for these spaces. An example of a visualization of two-dimensional samples and visual detection of outliers is given in the following figure.

OUTLIER ANALYSIS



OUTLIER ANALYSIS

- Quartiles, outliers and boxplots
  - Quartiles: Q_1 (25th percentile), Q_3 (75th percentile)
  - Inter-quartile range: IQR = $Q_3 - Q_1$
  - Five number summary: min, Q_1, M, Q_3, max
  - Boxplot: ends of the box are the quartiles, median is marked, whiskers, and plot outlier individually
  - Outlier: usually, a value higher/lower than 1.5 x IQR
  - A common rule for identifying suspected outliers is to single out values falling at least 1.5xIQR above the third quartile or below the first quartile.

OUTLIER ANALYSIS
BoxPlots

- Five-number summary of a distribution:

  Minimum, Q1, M, Q3, Maximum

- Boxplot
  - Data is represented with a box
  - The ends of the box are at the first and third quartiles, i.e., the height of the box is IRQ
  - The median is marked by a line within the box
  - Whiskers: two lines outside the box extend to Minimum and Maximum

OUTLIER ANALYSIS
BoxPlots